

MC-INR: Efficient Encoding of Multivariate Scientific Simulation Data using Meta-Learning and Clustered Implicit Neural Representations

Hyunsoo Son *
Korea University

Jeonghyun Noh †
Korea University

Suemin Jeon ‡
Korea University

Chaoli Wang §
University of Notre Dame

Won-Ki Jeong ¶
Korea University

ABSTRACT

Implicit Neural Representations (INRs) are widely used to encode data as continuous functions, enabling the visualization of large-scale multivariate scientific simulation data with reduced memory usage. However, existing INR-based methods face three main limitations: (1) inflexible representation of complex structures, (2) primarily focusing on single-variable data, and (3) dependence on structured grids. Thus, their performance degrades when applied to complex real-world datasets. To address these limitations, we propose a novel neural network-based framework, MC-INR, which handles multivariate data on unstructured grids. It combines meta-learning and clustering to enable flexible encoding of complex structures. To further improve performance, we introduce a residual-based dynamic re-clustering mechanism that adaptively partitions clusters based on local error. We also propose a branched layer to leverage multivariate data through independent branches simultaneously. Experimental results demonstrate that MC-INR outperforms existing methods on scientific data encoding tasks.

Index Terms: Implicit neural representation, meta-learning, clustering, multivariate data encoding.

1 INTRODUCTION

Scientific simulations numerically model spatiotemporal variations to capture complex physical phenomena. The resulting data typically consists of multiple variables or features, such as temperature, pressure, velocity, etc., that vary continuously over spatiotemporal coordinates. Visualizing such multivariate scientific data is essential to recognize patterns and outliers at particular spatial or temporal locations, which has a long-standing history of research [4]. Traditionally, multivariate scientific data have been represented using grids or polygonal meshes [10, 13, 20, 24]. However, recent neural network-based methods, such as implicit neural representations (INRs) [2, 3, 7, 8, 17, 18], have demonstrated remarkable performance, particularly in data compression and super-resolution, effectively reducing data size. Notably, INRs are well-suited for scientific visualization, as they approximate values at given spatiotemporal coordinates and represent them as continuous functions.

However, conventional INRs face several limitations in representing multivariate scientific simulation data. First, most existing INR methods rely on a single network to represent the entire domain, which restricts their ability to capture complex spatial structures and shapes. Second, they are primarily designed to handle single-variable data, limiting their ability to represent multivariate scientific simulations effectively. In real-world scientific simulation data, multiple physical variables interact within the same spatial do-

main, making it challenging for traditional representations to capture such complexity, often resulting in poor performance. Lastly, INRs for representing time-varying scientific data are generally biased toward structured grids, making it challenging to represent complex geometric structures in unstructured grids. As a result, they often fail to capture intricate spatial structures accurately.

To address these limitations, we propose **MC-INR** (*i.e.*, meta-learning and clustered INR), a novel neural network-based representation that extends CoordNet [2] for efficient encoding of multivariate data defined on unstructured grids. Our contributions are as follows: First, we combine meta-learning [22] and spatial clustering to enable efficient encoding of local patterns and flexible adaptation to complex structures. MC-INR partitions the data into spatial clusters and learns cluster-specific meta-knowledge. We also introduce a residual-based dynamic re-clustering mechanism that adaptively partitions clusters to more accurately capture structural variations with reduced error. Second, we propose a branched network architecture inspired by multi-task learning, in which each variable is assigned a dedicated branch to capture fine-grained structures, thereby improving the representation accuracy of multivariate data. Notably, the proposed network and spatial domain decomposition make our method particularly well-suited for representing multivariate data on unstructured grids. We demonstrate the effectiveness of MC-INR both quantitatively and qualitatively using our in-house small modular reactor simulation datasets.

2 RELATED WORK

2.1 Meta Learning

Meta-learning [1] is a training framework designed to enable models to quickly adapt to new tasks and complex structures. In particular, it has been widely applied in INRs to reduce training time and improve generalization performance. Sitzmann *et al.* [14] proposed a gradient-based meta-learning approach, and Tancik *et al.* [16] introduced a method that adaptively targets classes of signals, leveraging the benefits of meta-learning. Yang *et al.* [22] proposed Meta-INR, which improves the generalization and representation of time-varying data by interval-selecting points for training the meta-learner. Inspired by these strategies, we adopt meta-learning to enable efficient training and represent complex structures.

2.2 Clustering

Several scientific simulation datasets are large in scale, making it difficult to fully encode and utilize their complex features due to high computational memory requirements. To mitigate this limitation, various methods employed tree-based [9, 23] or Laplacian pyramid-based [12, 17] approaches to partition the data into smaller, more manageable clusters. Liu *et al.* [6] used k-means clustering to apply unstructured grids, addressing cases that previous methods find challenging to handle. Inspired by this, we employ the k-means clustering to treat unstructured grids; furthermore, each cluster is processed on a separate GPU to improve training speed and computational efficiency.

2.3 INR for Scientific Simulation Data

Traditionally, spatial data structures, such as rectilinear grids and polygonal meshes, have been used for scientific data visualiza-

*e-mail: ehsoon@korea.ac.kr

†e-mail: wjdus0967@korea.ac.kr

‡e-mail: orangeblush@korea.ac.kr

§e-mail: chaoli.wang@nd.edu

¶e-mail: wkjeong@korea.ac.kr

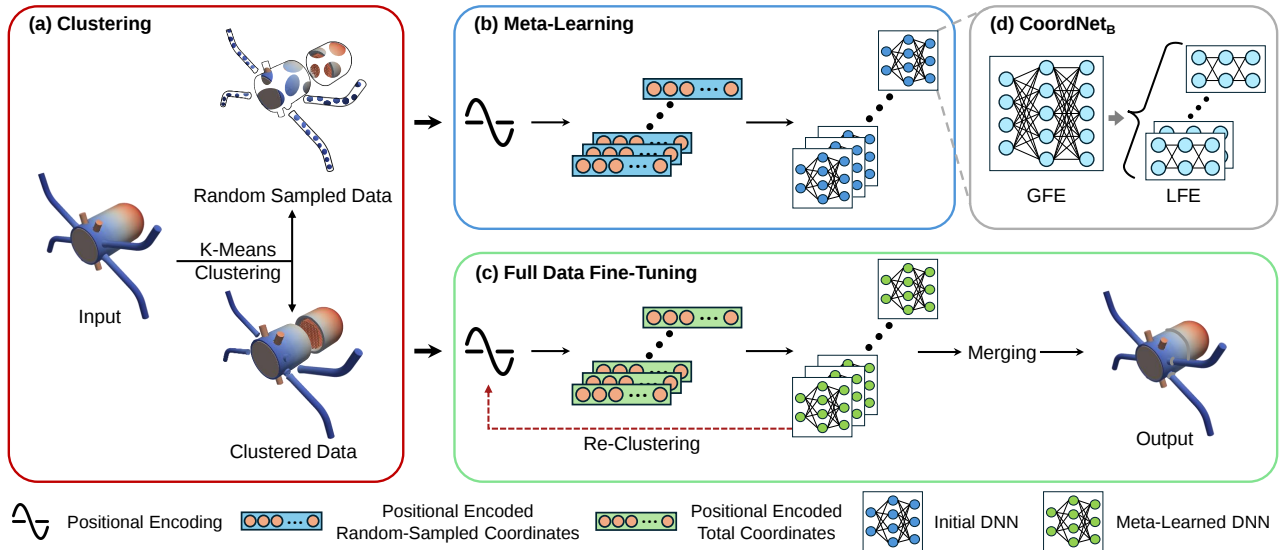


Figure 1: The overview of MC-INR: (a) The input data is spatially partitioned using k-means clustering, and random sampling is performed within each cluster for meta-learning. (b) The sampled data are passed through positional encoding and used to train an initial network via meta-learning. (c) The meta-learned model is then fine-tuned using the full data of each cluster. Re-clustering is performed if the residual exceeds a threshold. (d) The proposed CoordNet_B is the backbone DNN model, consisting of global and local structure feature extractors (GFE and LFE). Finally, the outputs from all clusters are merged to produce the final encoding result.

tion [10, 13, 20, 24]. Recently, numerous studies have emerged using INR [15], representing data as continuous functions. A pioneering work by Han and Wang [2] proposed coordinate-based data encoding using a neural network, *CoordNet*, improving the model’s representation performance on time-varying data. Wurster *et al.* [21] suggested adaptively placed feature grids in the encoder to effectively capture spatial characteristics. Tang and Wang [18] and Jiao *et al.* [3] introduced the INR methods based on deep neural networks (DNN) to handle the spatial and temporal variables. Wang *et al.* [19] proposed a hash encoder [11]-based method combined with finite differences to capture global and local information and improve boundary conditions. While prior works mainly focus on single-variable data in structured grids, limiting fine structure representation, we address this by exploring multivariate data on unstructured grids using a branched layer.

3 PROPOSED METHOD: MC-INR

MC-INR applies k-means clustering to partition the data and performs meta-learning using sampled points from each cluster. Each cluster is then trained using all its data points, guided by the acquired meta-knowledge. To handle multivariate data, MC-INR employs a branched network layer. The overview of MC-INR is shown in Fig. 1, and the detailed methodology is described below.

3.1 Meta Learning and Clustering

Directly training on large datasets can make it difficult to effectively encode the data, as the underlying characteristics may be complex and hard to capture. To address this, we adopt the k-means clustering strategy inspired by Liu *et al.* [6]. This method partitions the dataset, spatially coherent sub-regions based on spatial coordinates, preserving local structures and enabling the model to capture localized temporal variations. The cluster partition is defined as:

$$C = \arg \min_C \sum_{i=1}^K \sum_{\mathbf{x} \in C_i} \|\mathbf{x} - \mu_i\|^2, \quad (1)$$

where $C = \{C_1, C_2, \dots, C_K\}$ represents the set of K clusters, $\mathbf{x}$ denotes a data point which is spatial coordinate, and μ_i is the cen-

teroid of cluster C_i . The number of clusters, denoted as K , is set to 20. Each cluster is assigned an independent DNN and process, enabling concurrent execution across multiple GPUs for efficient resource utilization and improved training throughput.

To enhance the model’s performance, we further introduce a residual-based dynamic re-clustering mechanism. After training the initial cluster, we compute the residual by the mean squared error (MSE). If the residual of the cluster exceeds a threshold $\tau = 5e-4$, the cluster is subdivided into two sub-clusters to preserve local temporal variations. Otherwise, the training process is terminated. Each sub-cluster inherits the model parameters from the cluster before division, preserving learned knowledge while refining representations in high-error regions. Note that each cluster is trained by an independent DNN. With this strategy, our model effectively captures localized temporal variations.

INRs typically struggle to represent complex structures accurately; to mitigate this, we employ meta-learning [1], aiming to adapt to the complexities of scientific data. Inspired by Meta-INR [22], which performs sub-sampling by selecting points at regular intervals along each axis in structured grids, we extend this idea to unstructured grids. However, since unstructured data lacks regularity, interval-based sampling is not feasible. Therefore, we adopt a random sampling strategy, selecting points across the domain without relying on fixed intervals. This approach allows us to efficiently construct representative sub-sampled datasets suitable for meta-learning in unstructured grids. These meta-learning and clustering approaches are illustrated in Fig. 1 (a), (b), and (c).

3.2 Branched CoordNet

Scientific simulation data typically includes multiple physical variables; however, most existing methods are designed to handle only a single variable, limiting to represent multivariate datasets effectively. To address this issue, we propose a branched layer network based on CoordNet [2]. It consists of SIREN [15] network with residual blocks, which has deeper network depth for accurate encoding. Our network architecture is an extension of CoordNet, referred to as Branched CoordNet (CoordNet_B), consisting of global and local structure feature extractors (GFE and LFE). The GFE cap-

tures global structural features, such as overall patterns, while the LFE is designed to learn local features specific to each physical variable, including fine-grained details. Furthermore, to facilitate stable training and enhance the learning of fine-grained details, we leverage sine activation functions [15] along with skip connections within the network architecture. It is illustrated in Fig. 1 (d) and is formulated as follows:

$$\begin{aligned} \mathbf{h}_0 &= \mathbf{PE}([x, y, z, t]), \\ \mathbf{y}_j &= \mathbf{LFE}_j(\mathbf{GFE}(\mathbf{h}_0)), \quad j \in \{1, 2, \dots, M\}, \end{aligned} \quad (2)$$

where $\mathbf{h}_0$ and $\mathbf{PE}$ denote the input and positional encoding, respectively. x, y, z, t refers to the coordinate at time step t . $\mathbf{GFE}$ represents the GFE and $\mathbf{LFE}_j$ denotes the j -th LFE, where M is the total number of variables (e.g., temperature, pressure, etc), and $\mathbf{y}_j$ is the j -th output variable from the network. The feature extractor is formulated as follows:

$$\mathbf{FE}_l = \psi(\mathbf{W}_l \mathbf{h}_{l-1} + \mathbf{b}_l), \quad l \in \{1, 2, \dots, N\}, \quad (3)$$

where $\mathbf{FE}_l$ represents the l -th feature extractor layer consisting of the $\mathbf{GFE}$ and $\mathbf{LFE}_j$, where N is number of layers. $\mathbf{h}_{l-1}$, ψ , $\mathbf{W}_l$ and $\mathbf{b}_l$ correspond to feature map, sine activation function, weight, and bias at the l -th layer, respectively. Training is conducted independently for each cluster C_i , and to optimize each cluster, we employ MSE as a loss function. By leveraging branched layers within the network, this approach further facilitates efficient joint learning of multiple variables within a single model.

4 EXPERIMENTS

4.1 Implementation Details

Datasets. We used a variety of multivariate scientific simulation datasets, including in-house datasets, ROCOM, CUPID, and synthetic datasets provided by our collaborator. Each dataset is defined on an unstructured tetrahedron mesh. The dataset information, used in our experiments, is presented in Tab. 1.

Implementation. We conducted experiments using PyTorch 2.1.0 on four NVIDIA RTX A6000 GPUs. The meta-learning training hyperparameters are set to be the same as those used in Meta-INR [22]. The network consists of a total of 11 layers, comprising 5 layers in the GFE and 6 layers in the LFE, where each layer is implemented using the residual block design introduced in CoordNet [2]. The width of the network was adjusted to the size and characteristics of each dataset. We performed the main training until convergence using the Adam optimizer with an initial learning rate of $5e-5$. The convergence condition is defined as no decrease in training loss over 30 epochs. A learning rate scheduler is applied, reducing the learning rate by a factor of 0.92 every 30 epochs. Note that we used 30% of the total points that were randomly sampled in each epoch during the main training. We used peak signal-to-noise ratio (PSNR), normalized root mean squared error (NRMSE), and coefficient determination (R^2) for quantitative evaluation. For the quantitative comparison, we average the result values across all variables and timesteps. The bold text indicates the best result.

4.2 Results

Quantitative results. Tab. 2 presents the experimental results comparing with previous methods across various datasets. We selected baseline methods that are independent of grid structure and can be adapted to support multivariate outputs, ensuring a fair comparison with our approach. MC-INR consistently outperforms prior methods, demonstrating superior accuracy. Specifically, without meta-learning and clustering, our network architecture, CoordNet_B, achieves better performance than SIREN [15] and CoordNet across all three metrics, highlighting the effectiveness of the branched layer for multivariate data. By integrating meta-learning and clustering, MC-INR further improves performance, achieving superior

Table 1: Experimental datasets overview. p and t denote the number of spatial coordinates per timestep and the number of timesteps, respectively.

Dataset	# of points ($p \times t$)	# of variables
ROCOM	12,440,366 $\times$ 5	8
CUPID	1,279,375 $\times$ 30	11
Synthetic	5,253,750 $\times$ 30	4

Table 2: Quantitative comparison of data encoding results. Size indicates the model's weight, and for our method, it refers to the total size summed over all clusters. CoordNet_B means our branched network, which is without meta-learning and clustering.

Dataset	Method	PSNR $\uparrow$	NRMSE $\downarrow$	$R^2\uparrow$	Size (KB)
ROCOM	SIREN [15]	38.59	0.0161	0.8756	530
	CoordNet [2]	39.51	0.0129	0.8133	23215
	CoordNet _B	47.65	0.0073	0.9765	10372
	MC-INR	54.10	0.0042	0.9923	33600
CUPID	SIREN	21.63	0.1656	0.5659	304
	CoordNet	20.51	0.1732	0.4733	13095
	CoordNet _B	36.99	0.0160	0.8294	6231
	MC-INR	51.55	0.0033	0.9975	27240
Synthetic	SIREN	52.04	0.0039	-0.3349	528
	CoordNet	50.39	0.0038	-8.4193	23199
	CoordNet _B	62.16	0.0010	0.7958	9627
	MC-INR	76.54	0.0002	0.9827	30180

results across all metrics. This improvement stems from its ability to efficiently capture local patterns and complex structures. We observed that SIREN and CoordNet yield negative R^2 scores on the Synthetic dataset, due to an outlier in one of the output channels. We suspect this is a limitation of multichannel vector regression using DNN, though further investigation is needed. Although our method requires the largest model weight size, we observed that baseline models with similar sizes still show lower performance compared to ours, making our approach the only viable option for reducing encoding error. To further demonstrate training efficiency, we present the PSNR results over training time in Fig. 2. Within the same training duration, MC-INR has achieved a higher PSNR than other methods, which indicates that MC-INR is effective and efficient for multivariate data encoding.

Compression results. To demonstrate the compression performance of MC-INR's, we compare PSNR at different compression ratios (CR, raw data size / model's weight size) in Tab. 3. We compared our method to SZ3 [5], commonly used for scientific data compression that achieves high CR while preserving user-defined error bounds. For a fair comparison, we evaluate SZ3 and MC-INR under two aligned conditions: one with similar CRs to assess differences in PSNRs, and the other with similar PSNRs to assess differences in CRs. As shown in Tab. 3, MC-INR achieved better performance in both cases, yielding higher PSNR at similar CRs and higher CR at similar PSNRs.

Qualitative results. Visual comparison with previous methods is

Table 3: Quantitative comparison of data compression results. We evaluated under identical PSNR and CR settings to ensure a fair comparison. CR refers to the compression ratio.

Dataset	Method	PSNR $\uparrow$	CR $\uparrow$
ROCOM	SZ3 [5]	22.83	109
	SZ3	54.07	4.89
	MC-INR	54.10	110
CUPID	SZ3	16.69	229
	SZ3	51.30	2.03
	MC-INR	51.55	230
Synthetic	SZ3	36.96	310
	SZ3	76.05	14.61
	MC-INR	76.54	311

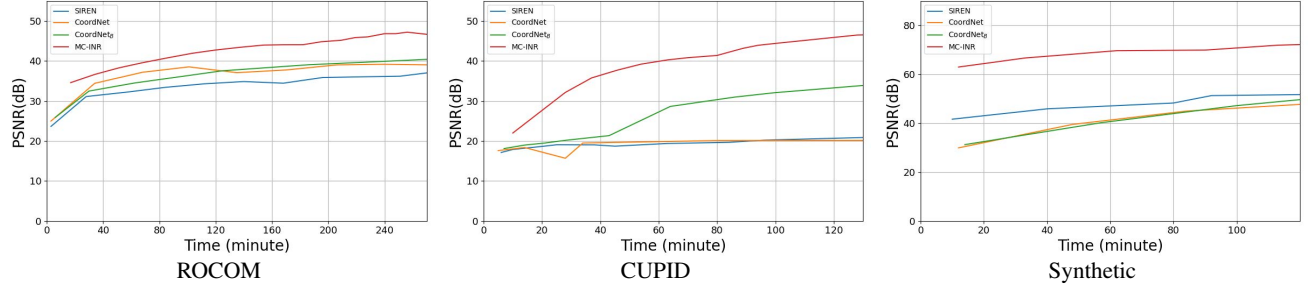


Figure 2: Comparison of encoding performance over time. PSNR is recorded per epoch, with curves starting at different times due to varying training durations. Each method is trained until convergence.

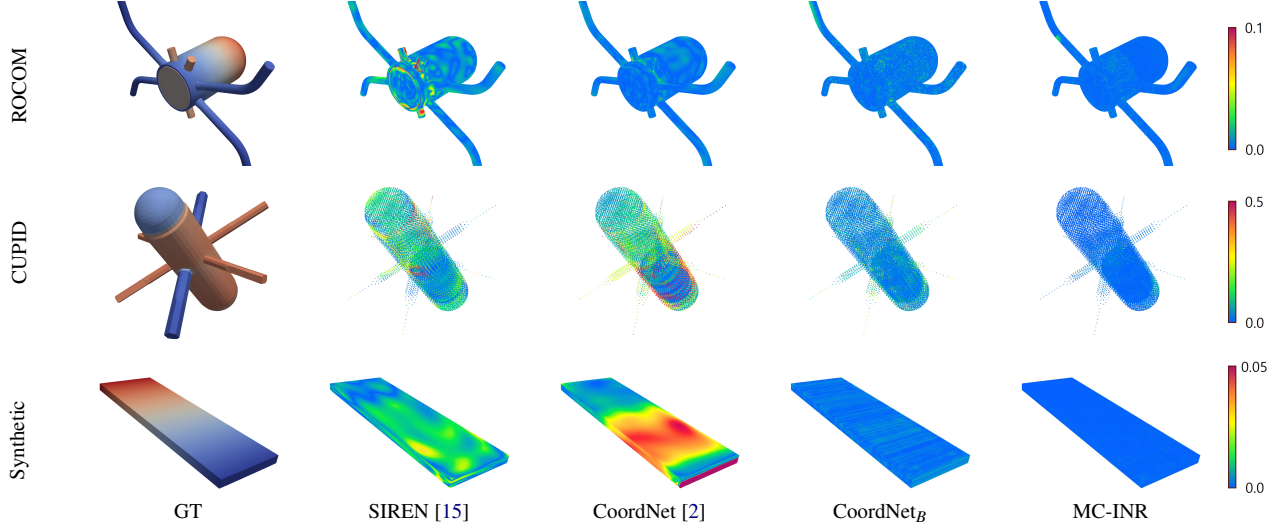


Figure 3: Error map visualization of pressure value. Errors are computed using absolute difference, with lower values shown in blue and higher values in red. The ground truth (GT) refers to the reconstructed points obtained from the tetrahedral mesh.

Table 4: Ablation study results for residual-based re-clustering. This experiment is conducted on the ROCOM dataset.

Method	PSNR $\uparrow$	NRMSE $\downarrow$	R ² $\uparrow$
w/o Re-clustering	51.51	0.0052	0.9886
w/ Re-clustering	54.10	0.0042	0.9923

illustrated in Fig. 3. The ground truth (GT) rendered multivariate datasets by a tetrahedral mesh, and other methods visualized the error map illustrating the absolute difference between the GT and the prediction, with lower errors shown in blue and higher errors in red, highlighting regions of inaccuracy. SIREN and CoordNet exhibit high errors on the synthetic data, while CoordNet_B exhibits significantly lower errors compared to them. MC-INR further reduces the error of CoordNet_B, demonstrating its effectiveness in encoding multivariate datasets.

Ablation study. We evaluate the effectiveness of the residual-based dynamic re-clustering mechanism on the ROCOM dataset (Tab. 4). When re-clustering is applied, the model achieves a 2.59 dB improvement in PSNR compared to without re-clustering, along with slight gains in the other two metrics. This improvement indicates that re-clustering enables the model to enhance the capture of specific local structures. Therefore, the re-clustering mechanism is effective in improving the model’s representational ability.

4.3 Limitations

Due to the use of multiple networks (one per cluster), our method incurs higher memory consumption compared to approaches using a single network. Additionally, non-overlapping clusters may introduce boundary artifacts, which can become noticeable when zooming into local regions near cluster edges. Introducing simple boundary overlaps could mitigate this issue. Despite these limitations, our method achieves superior encoding performance compared to previous works.

5 CONCLUSION

In this paper, we introduced MC-INR, a novel neural network-based framework designed to encode multivariate scientific simulation data on unstructured grids. MC-INR leverages meta-learning and clustering strategies to effectively encode data on complex unstructured grids and capture local patterns. The proposed residual-based dynamic re-clustering mechanism further contributes to improving the encoding accuracy. Additionally, we introduced CoordNet_B, which enables joint learning of multivariate data and enhances representation quality. As a result, MC-INR achieved superior performance in both quantitative and qualitative evaluations, demonstrating its effectiveness and robustness for scientific data visualization.

For future work, we plan to explore various clustering strategies that leverage both spatial structures and data distributions for higher encoding performance. Moreover, we intend to conduct in-depth validation across diverse datasets, including both structured and unstructured grid data.

ACKNOWLEDGMENTS

This work was partially supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2024-00349697), the National Research Council of Science & Technology (NST) grant by MSIT (No. GTL24031-000), the ICT Creative Consilience program of the Institute for Information & communications Technology Planning & Evaluation (IITP) funded by MSIT (IITP-2025-RS-2020-II201819), and a Korea University Grant.

REFERENCES

- [1] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proc. of Int'l Conf. on Machine Learning (ICML)*, pp. 1126–1135, 2017. 1, 2
- [2] J. Han and C. Wang. Coordnet: Data generation and visualization generation for time-varying volumes via a coordinate-based neural network. *IEEE Trans. on Visualization and Computer Graphics (TVCG)*, 29(12):4951–4963, 2022. 1, 2, 3, 4
- [3] C. Jiao, C. Bi, and L. Yang. FFEINR: Flow feature-enhanced implicit neural representation for spatiotemporal super-resolution. *Journal of Visualization*, 27(2):273–289, 2024. 1, 2
- [4] K. Lachmann, J. Kehler, J. Kehler, and H. Hauser. Visualization and visual analysis of multifaceted scientific data. *IEEE Trans. on Visualization and Computer Graphics (TVCG)*, 18(12):2324–2338, 2012. 1
- [5] X. Liang, K. Zhao, S. Di, S. Li, R. Underwood, A. M. Gok, J. Tian, J. Deng, J. C. Calhoun, D. Tao, et al. SZ3: A modular framework for composing prediction-based error-bounded lossy compressors. *IEEE Transactions on Big Data*, 9(2):485–498, 2022. 3
- [6] K. Liu, C. Jiao, X. Gao, and C. Bi. UGINR: Large-scale unstructured grid reduction via implicit neural representation. *Journal of Visualization*, 27(5):983–996, 2024. 1, 2
- [7] Y. Lu, P. Gu, and C. Wang. FCNR: Fast compressive neural representation of visualization images. In *IEEE Visualization and Visual Analytics (VIS)*, pp. 31–35, 2024. 1
- [8] Y. Lu, K. Jiang, J. A. Levine, and M. Berger. Compressive neural representations of volumetric scalar fields. In *Computer Graphics Forum*, vol. 40, pp. 135–146, 2021. 1
- [9] J. N. Martel, D. B. Lindell, C. Z. Lin, E. R. Chan, M. Monteiro, and G. Wetzstein. Acorn: Adaptive coordinate networks for neural scene representation. *arXiv preprint arXiv:2105.02788*, 2021. 1
- [10] P. Muigg, M. Hadwiger, H. Doleisch, and H. Hauser. Scalable hybrid unstructured and structured grid raycasting. *IEEE Trans. on Visualization and Computer Graphics (TVCG)*, 13(6):1592–1599, 2007. 1, 2
- [11] T. Müller, A. Evans, C. Schied, and A. Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. on Graph. (ToG)*, 41(4):1–15, 2022. 2
- [12] V. Saragadam, J. Tan, G. Balakrishnan, R. G. Baraniuk, and A. Veeraraghavan. Miner: Multiscale implicit neural representation. In *Proc. of European Conf. on Computer Vision (ECCV)*, pp. 318–333, 2022. 1
- [13] H.-W. Shen. Visualization of large scale time-varying scientific data. In *Journal of Physics: Conference Series*, vol. 46, p. 535. IOP Publishing, 2006. 1, 2
- [14] V. Sitzmann, E. Chan, R. Tucker, N. Snavely, and G. Wetzstein. Metasdf: Meta-learning signed distance functions. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:10136–10147, 2020. 1
- [15] V. Sitzmann, J. Martel, A. Bergman, D. Lindell, and G. Wetzstein. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:7462–7473, 2020. 2, 3, 4
- [16] M. Tancik, B. Mildenhall, T. Wang, D. Schmidt, P. P. Srinivasan, J. T. Barron, and R. Ng. Learned initializations for optimizing coordinate-based neural representations. In *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 2846–2855, 2021. 1
- [17] K. Tang and C. Wang. ECNR: Efficient compressive neural representation of time-varying volumetric datasets. *arXiv preprint arXiv:2311.12831*, 2023. 1
- [18] K. Tang and C. Wang. STSR-INR: Spatiotemporal super-resolution for multivariate time-varying volumetric data via implicit neural representation. *Computers & Graphics*, 119:103874, 2024. 1, 2
- [19] H. Wang, T. Yu, T. Yang, H. Qiao, and Q. Dai. Neural physical simulation with multi-resolution hash grid encoding. In *Proc. of AAAI Conf. on Artificial Intelligence (AAAI)*, vol. 38, pp. 5410–5418, 2024. 2
- [20] J. Wang, S. Hazarika, C. Li, and H.-W. Shen. Visualization and visual analysis of ensemble data: A survey. *IEEE transactions on visualization and computer graphics*, 25(9):2853–2872, 2018. 1, 2
- [21] S. W. Wurster, T. Xiong, H.-W. Shen, H. Guo, and T. Peterka. Adaptively placed multi-grid scene representation networks for large-scale data visualization. *IEEE Trans. on Visualization and Computer Graphics (TVCG)*, 30(1):965–974, 2023. 2
- [22] M. Yang, K. Tang, and C. Wang. Meta-INR: Efficient encoding of volumetric data via meta-learning implicit neural representation. *arXiv preprint arXiv:2502.09669*, 2025. 1, 2, 3
- [23] R. Yang, T. Xiao, Y. Cheng, Q. Cao, J. Qu, J. Suo, and Q. Dai. Sci: A spectrum concentrated implicit neural compression for biomedical data. In *Proc. of AAAI Conf. on Artificial Intelligence (AAAI)*, vol. 37, pp. 4774–4782, 2023. 1
- [24] H. Yu, C. Wang, R. W. Grout, J. H. Chen, and K.-L. Ma. In situ visualization for large-scale combustion simulations. *IEEE computer graphics and applications*, 30(3):45–57, 2010. 1, 2